Fulton Hogan Proposal

Semantic Model Refresh Framework v2.0

Date: February 2026

Executive Summary

Fulton Hogan's Enterprise Data Platform supports multiple Power BI semantic models across AWM, Finance, and Operations, each with different freshness requirements. The proposed framework establishes a single platform pattern for semantic refresh orchestration on Databricks Jobs using metadata-driven policy, model-level isolation, and operational observability.

1) The Problem

The current operating model creates avoidable risk in freshness, trust, and platform efficiency.

- Inconsistent freshness behavior across models using shared Gold tables.
- No single system of record for model SLA thresholds and refresh policy.
- Shared workflow coupling where one model failure can delay other models.
- Limited visibility into refresh health, SLA breaches, and cost by model.
- Manual operational overhead to answer "is this dashboard current?"

Business impact:

- Decision lag for near-real-time domains.
- Reduced trust in reporting outputs.
- Increased compute spend from inefficient refresh patterns.
- Higher support load for platform and domain teams.

2) The Solution

Implement a metadata-driven semantic refresh framework on Databricks with model-based workflow isolation and native Power BI task execution.

Design choices:

- Model-level refresh (one Power BI refresh per semantic model).
- One workflow per model per cadence.
- Native Databricks table update triggers for NRT scenarios.
- SLA thresholds managed as data in Unity Catalog metadata.
- Native Databricks Power BI task as the standard refresh task.
- Unified logging and cost tracking for each refresh execution.

Expected outcomes:

• Stronger SLA compliance and predictable refresh behavior.

• Failure isolation across models.

• Reduced operational noise and faster root-cause analysis.

• Clear cost and health visibility for governance.

3) Solution Components (High Level)

A. Metadata and Governance Layer

Unity Catalog schema:

• `edp_metadata.semantic_refresh`

Core Delta tables:

• `semantic_models`

• `table_refresh_config`

• `refresh_execution_log`

• `refresh_cost_tracking`

Purpose:

• Defines what to refresh, when to refresh, and what SLA applies.

• Captures complete operational and cost audit trails.

B. Model-Oriented Workflow Orchestration

Workflow pattern:

• One workflow per model per cadence (e.g., `AWM-DAILY`, `Finance-MEDIUM_FREQUENCY`, `AWM-TABLE_TRIGGERED`).

Standard job sequence:

1. `log_start`
2. `refresh_model` (Power BI task)
3. `log_complete`
4. `capture_costs`
5. `on_failure` notifications

Purpose:

• Ensures fault isolation, cleaner ownership, and SLA-specific tuning.

C. Trigger and Refresh Execution Layer

Trigger modes:

• Scheduled cadence for daily/medium frequency.

• Native `table_update` triggers for NRT source changes with debounce.

Refresh execution:

• Databricks Power BI task using `connection_name`, `dataset_id`, `refresh_type`.

• Refresh scope remains semantic-model level.

Purpose:

• Aligns execution to business freshness requirements while minimizing waste.

D. Monitoring, SLA, and Cost Intelligence

Operational scorecard inputs:

• Success rate (7-day), SLA breaches, duration, staleness, cost per refresh.

Delivery surface:

• Databricks SQL health dashboard over execution/cost metadata.

Purpose:

• Creates a single trusted view for platform health and executive governance.

4) Implementation Scope

Core files:

1. `01_create_metadata_schema.sql`
2. `02_grant_permissions.sql`
3. `03_create_powerbi_connections.sql`
4. `04_seed_metadata.sql`
5. `05_generate_workflows_model_based.py`
6. `metadata_helper.py`
7. `log_model_refresh.py`
8. `capture_refresh_costs.py`
9. `sla_monitoring.py`
10. `dashboard_queries.sql`
11. `model_health_report.sql`

Deployment phases:

• Pilot: establish metadata + pilot workflows.

• Scale: expand to all active models and cadences.

• NRT: enable validated table update triggers for approved domains.

5) High-Level Picture

1. Gold domain tables in Unity Catalog produce freshness signals.
2. Metadata defines per-model policy, cadence, trigger mode, and SLA.
3. Workflow generator creates model-isolated Databricks Jobs.
4. Each job runs Power BI refresh with start/complete/cost telemetry.
5. SQL dashboards provide health, SLA, and cost transparency.

Final result:

• A consistent, auditable, and scalable semantic refresh operating model for Fulton Hogan that improves trust, reliability, and cost control.